National Space Day Hackathon 2025

Department of Physics, Guru Nanak Dev University, Amritsar

Problem Statement Booklet

Problem: Chandrayaan-4 Lunar Outpost Mandate

Event Date: August 20, 2025

Organized in celebration of India's historic Chandrayaan-3 landing.

Event Schedule & Welcome

Welcome, future scientists and engineers, to the National Space Day Hackathon 2025! Today, we commemorate India's incredible achievements in space exploration by challenging you with a problem that reflects the real-world complexities faced by our nation's space program.

Your mission is not just about coding; it's about analytical thinking, strategic decision-making, and scientific rigor. Good luck!

Hackathon Timeline (3 Hours Total)

Event Flow

- **00:00 – 00:15 (15 mins): Briefing & Setup**
  Problem statement distribution. Teams may set up their development environments. The datasets will be made available via the provided link.
- **00:15 – 02:45 (2.5 hours): Coding Phase**
  The main hacking period. Teams will analyze the datasets, develop their algorithms, and determine the optimal landing site.
- **02:45 – 03:00 (15 mins): Final Submissions**
  Teams must finalize their code and submit their solution. Late submissions will not be accepted.
- **Post-Hackathon: Judging & Results**
  The judging panel will evaluate all submissions based on the official marking scheme. Results will be announced later in the evening.

Dataset Access

The required '.csv' data files can be downloaded from the following repository:

https://github.com/GNDU-Physics/NSpD25

Your code must be pushed in the same repository within last 15 minutes with your code stored in your team name folder!

1 Problem Statement: The Chandrayaan-4 Lunar Outpost Mandate

1.1 Mission Directive

Official Communiqué

To: Mission Operations Complex (MOX), ISTRAC
From: Chandrayaan-4 Mission Directorate
Date: August 23, 2025
Subject: Critical Mandate: Optimal Landing Site Selection for India's First Lunar Research Outpost.

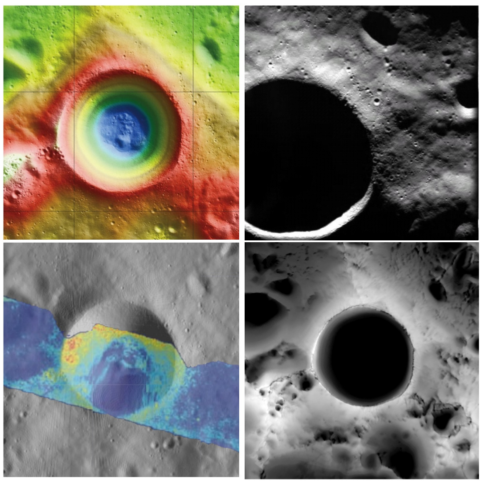
1.2 Introduction

Following the monumental successes of the Chandrayaan program, the next national objective is the establishment of a semi-permanent, long-duration research outpost on the Moon. The **Chandrayaan-4** mission will be the cornerstone of this effort, tasked with landing the primary habitat and power infrastructure.

The mission's success is critically dependent on the selection of the optimal landing site. Data from the Lunar Reconnaissance Orbiter (LRO) and previous Chandrayaan missions has given us unprecedented insight into the lunar south pole—a region of immense scientific and strategic value. However, it is also a region of extreme environmental contrasts.

1.3 The Challenge

Your team's objective is to analyze a multi-layered geospatial dataset for a candidate 50km x 50km region near the Shackleton crater. You must identify the single optimal **1km x 1km** grid cell for the Chandrayaan-4 landing and initial outpost construction.



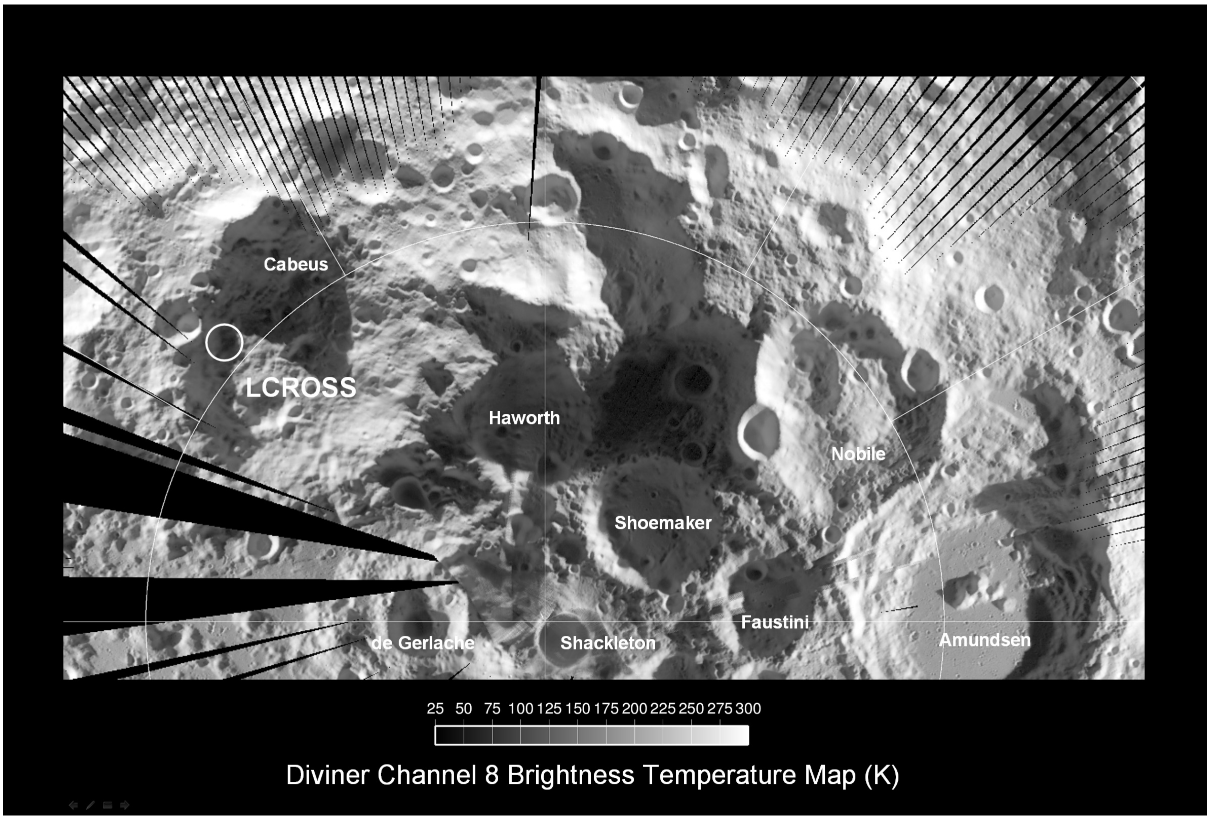
Shackleton Crater at the lunar south pole — potential site for Chandrayaan-4 outpost.

The "optimal" site is not merely the flattest point. It is a complex trade-off between engineering safety, resource availability, and power generation capability. **Your task is to write a program to process this data and justify your selection based on a quantitative scoring model.**

1.4 Mission Parameters & Datasets

You will be provided with four CSV files, each representing a data layer for the 50km x 50km candidate region. The region is represented as a `500x500` grid, where each cell is `100m x 100m`. Your target landing zone is a `10x10` cell area (i.e., `1km x 1km`).

1. `elevation.csv`: A `500x500` matrix of elevation values in meters.
2. `illumination.csv`: A `500x500` matrix where each value represents the average percentage of time that cell is illuminated by the sun over a full lunar year.
3. `water_ice.csv`: A `500x500` matrix of survey data representing the probability (from 0.0 to 1.0) of finding significant water-ice deposits.
4. `signal_occultation.csv`: A `500x500` matrix where each value is the total number of hours per year that the direct line-of-sight to Earth is blocked.



South lunar pole as imaged by the Diviner instrument on the NASA's Lunar Reconnaissance Orbiter. Shackleton is at bottom center.

2 Analytical Task & Final Objective

2.1 Your Analytical Task

Your program must perform the following analysis for **every possible** `1km x 1km` (`10x10` cell) square within the larger map:

1. **Terrain Roughness Score (S_{terrain}):** For each `10x10` square, calculate the standard deviation of the elevation values ($\sigma_{\text{elevation}}$). A lower standard deviation means a flatter, safer terrain. To make this a positive score where higher is better, use:

$$S_{\text{terrain}} = \frac{1}{1 + \sigma_{\text{elevation}}}$$

2. **Illumination Score (S_{light}):** Calculate the **average** illumination percentage within the `10x10` square.

3. **Water-Ice Score (S_{water}):** Calculate the **average** water-ice probability within the `10x10` square.

4. **Communication Score (S_{comm}):** Calculate the **average** signal occultation hours. Similar to terrain, invert it so higher is better:

$$S_{\text{comm}} = \frac{1}{1 + \text{avg\_occultation\_hours}}$$

2.2 Final Objective: The Suitability Index

To find the single best site, you must combine these scores into a unified **Suitability Index ($I_{\text{suitability}}$)**. The Mission Directorate has provided the following weights based on mission priorities:

- Safety & Communication (Terrain, Comms): **60%**
- Power (Illumination): **25%**
- Resource Potential (Water): **15%**

First, you must **normalize** each of your four calculated scores ($S_{\text{terrain}}, S_{\text{light}}, S_{\text{water}}, S_{\text{comm}}$) to a common scale (0 to 1) across all possible landing sites. Let these be $S'_{\text{terrain}}, S'_{\text{light}}, S'_{\text{water}}, S'_{\text{comm}}$.

The final Suitability Index for each potential site is calculated with the following formula:

$$I_{\text{suitability}} = (0.40 \cdot S'_{\text{terrain}}) + (0.20 \cdot S'_{\text{comm}}) + (0.25 \cdot S'_{\text{light}}) + (0.15 \cdot S'_{\text{water}})$$

Required Final Output

Your program's final submission must be a text file named `result.txt` containing:

1. The `(row, column)` coordinates of the top-left cell of the single best `1km x 1km` landing zone.
2. The final, highest **Suitability Index** value achieved, to 6 decimal places.
3. The four individual (non-normalized) scores for your chosen site: terrain roughness ($\sigma_{\text{elevation}}$), average illumination, average water-ice probability, and average occultation hours.

3 Official Marking Scheme

Total Points Available: 100

The final score for each team will be a sum of points from three main categories: **Functional Correctness**, **Code Quality**, and **Performance**.

3.1 Functional Correctness (70 Points)

This category evaluates whether your program works as intended and produces the correct results.

- **Final Output Accuracy (40 Points):**
  - Correct Landing Zone Coordinates `(row, col)`: **20 points.**
  - Correct Final Suitability Index ($I_{\text{suitability}}$): **10 points.**
  - Correct Individual Scores for the chosen site: **10 points.**
- **Algorithmic Logic (30 Points):** (For partial credit, assessed via code review)
  - Correct calculation of raw scores (S_{terrain}, etc.): **15 points.**
  - Correct normalization of scores to a 0-1 scale: **10 points.**
  - Correct implementation of the final weighted formula: **5 points.**

3.2 Code Quality & Implementation (20 Points)

This category assesses the craftsmanship of your solution.

- **Readability & Structure (10 Points):**
  - Well-structured code with functions for different tasks.
  - Clear and meaningful variable names.
  - Comments explaining complex parts of the logic.
- **Data Processing & Libraries (10 Points):**
  - Use of appropriate and efficient libraries for data manipulation (e.g., **NumPy** or **Pandas**) is highly encouraged and will score higher than solutions using native Python lists and loops for matrix operations.

3.3 Performance & Efficiency (10 Points)

This category rewards teams who solved the problem elegantly and efficiently.

- **Baseline (Brute-Force): (1-4 points)**
- **Good (Vectorized): (5-7 points)**
- **Excellent (Highly Optimized): (8-10 points)**
  Solutions using advanced algorithms like **2D convolution** or a **summed-area table (integral image)** to avoid redundant calculations will receive the highest marks in this category.

4 General Rules & Guidelines

1. **Team Size:** Teams can consist of 1 to 4 members.
2. **Permitted Languages:** Any programming language is permitted, as long as it can read CSV files and perform the required mathematical operations. Python, C++, Java, and R are common choices. Frontend languages can be used for making the output visually appealing.
3. **Internet Access:** Internet access is permitted for looking up documentation and libraries.
4. **Pre-written Code:** You may not use pre-written code that solves the core logic of this specific problem. You may use your own or public libraries for general tasks (e.g., data input, math functions).
5. **Submission:** Each team must submit a single ZIP file containing:
    - All source code files.
    - The final `result.txt` output file.
    - A `README.md` file explaining how to run your code.

    The ZIP / Folder must be pushed within last 15 minutes in the repository mentioned.
6. **Academic Integrity:** All work must be your own. **Use of AI/GPTs are strictly prohibited.** If caught, will result in immediate disqualification. The decision of the judges is final.

Hackathon Coordinator Contact

For any technical issues with data access or clarifications regarding the problem statement, please contact the event coordinator. Please note that coordinators cannot provide hints on how to solve the problem.

Jashanpreet Singh Dingra
Mad Astrophile, Department of Physics
Email: astrodingra@gmail.com
Mobile: +91-88472-42013

Good Luck, and Happy Hacking!